

Лабораторна робота №4

CRUD-операції в EF Core. Валідація даних.

Мета: набути навичок роботи з ORM, навчитися реалізовувати операції створення, читання, оновлення та видалення даних з використанням EF Core, набути навичок роботи з механізмами валідації даних.

Хід роботи:

Завдання 1. Використовуючи підхід Code First, створити модель, яка разом з існуючою моделлю, утворюватиме зв'язок один-до-багатьох.

Створення моделі для «Ticket»

```
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace BookingTickets.Models
{
    public class Ticket
    {
        [Key]
        public long TicketID { get; set; }

        public string Seat { get; set; } = string.Empty;

        [Column(TypeName = "decimal(8,2)")]
        public decimal Cost { get; set; }

        public long EventID { get; set; }

        public Event Event { get; set; }
    }
}
```

Додавання зв'язків до Event

```
public ICollection<Ticket> Tickets { get; set; } = new List<Ticket>();
```

Міграція :

```
PS G:\4Courses\ASP\Lab4\BookingTicketsSLN\BookingTickets> dotnet ef migrations add AddedCategoryEventRelation
Build started...
Build succeeded.
The Entity Framework tools version '8.0.0' is older than that of the runtime '8.0.6'. Update the tools for the latest features and bug fixes. See https://aka.ms/AAc1fbw for more information.
Done. To undo this action, use 'ef migrations remove'
PS G:\4Courses\ASP\Lab4\BookingTicketsSLN\BookingTickets> dotnet ef database update
Build started...
Build succeeded.
The Entity Framework tools version '8.0.0' is older than that of the runtime '8.0.6'. Update the tools for the latest features and bug fixes. See https://aka.ms/AAc1fbw for more information.
[Info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (16ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT 1]
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.01.000 – Пр.4		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.	Аліев О.Е.				Пім.	Арк.	Аркушів
Перевір.	Бейрак Д.Я.					1	
Реценз.							
Н. Контр.							
Зас.каф.					Звіт з лабораторної		
					роботи №4		
					ФІКТ, гр. ІПЗ-22-1		

Завдання 2. Реалізувати CRUD (Create, Read, Update, Delete) операції для обох моделей

```
using System.Linq;

namespace BookingTickets.Models
{
    public interface IBookingRepository
    {
        IQueryable<Event> Events { get; }
        IQueryable<Ticket> Tickets { get; }

        void CreateEvent(Event e);
        void UpdateEvent(Event e);
        void DeleteEvent(Event e);

        void CreateTicket(Ticket t);
        void UpdateTicket(Ticket t);
        void DeleteTicket(Ticket t);
    }

    public void CreateEvent(Event e)
    {
        context.Events.Add(e);
        context.SaveChanges();
    }

    public void UpdateEvent(Event e)
    {
        context.SaveChanges();
    }

    public void DeleteEvent(Event e)
    {
        context.Events.Remove(e);
        context.SaveChanges();
    }

    public void CreateTicket(Ticket t)
    {
        context.Tickets.Add(t);
        context.SaveChanges();
    }

    public void UpdateTicket(Ticket t)
    {
        context.SaveChanges();
    }

    public void DeleteTicket(Ticket t)
    {
        context.Tickets.Remove(t);
        context.SaveChanges();
    }
}
```

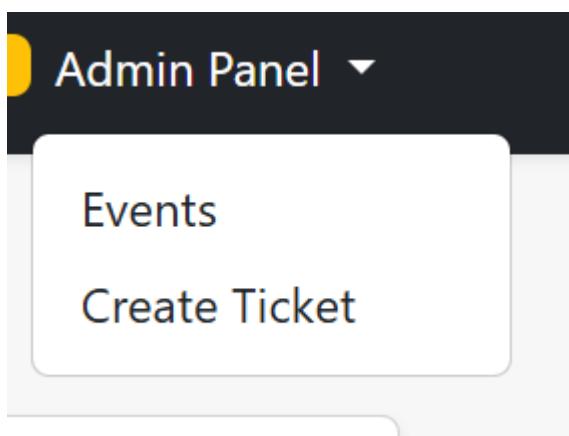
Змн.	Арк.	№ докум.	Підпис	Дата

Створив контролери і view для них :

```
C# EventController.cs  
C# HomeController.cs  
C# TicketController.cs
```

```
Event  
Create.cshtml  
Delete.cshtml  
Details.cshtml  
Edit.cshtml  
Index.cshtml
```

```
Ticket  
Create.cshtml  
Delete.cshtml  
Details.cshtml  
Edit.cshtml  
Index.cshtml
```



Title	Category	Price	
Rock Festival123	Music	640,00	Details Edit Delete
Stand-Up Night	Comedy	45,00	Details Edit Delete
Jazz Evening	Music	80,00	Details Edit Delete
Theatre Night	Art	150,00	Details Edit Delete
E-sports Championship	Gaming	95,00	Details Edit Delete
Hip-Hop Battle	Dance	60,00	Details Edit Delete
test1	test	1000,00	Details Edit Delete

Змн.	Арк.	№ докум.	Підпис	Дата	Арк.
					3

Create Event

Title

Description

Category

Price

Create

Tickets

Home

Cart

0

Admin Panel

Create Ticket

Seat

Cost

Event

Rock Festival123

Rock Festival123

Stand-Up Night

Jazz Evening

Theatre Night

E-sports Championship

Hip-Hop Battle

test1

Завдання 3. Додати серверну валідацію для перевірки введених користувачем даних перед виконанням CRUD-операцій.

Змн.	Арк.	№ докум.	Підпис	Дата

Art
Comedy
Dance
Gaming
Music
test4

Create Event

- The value 'test2' is not valid for Price.

Title: test2
Description: test2
Category: test2
Price: test2

The value 'test2' is not valid for Price.

Create

© 2025 BookingTickets

```

using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace BookingTickets.Models
{
    public class Event
    {
        [Key]
        public long EventID { get; set; }

        [Required(ErrorMessage = "Подія повинна мати назву.")]
        [StringLength(100, ErrorMessage = "Назва не може бути довшою за 100 символів.")]
        public string Title { get; set; } = string.Empty;

        [Required(ErrorMessage = "Опис теж потрібен.")]
        [StringLength(1000, ErrorMessage = "Опис занадто довгий.")]
        public string Description { get; set; } = string.Empty;

        [Required(ErrorMessage = "Вкажіть ціну.")]
        [Range(0.01, 999999, ErrorMessage = "Ціна має бути більшою за нуль.")]
        [Column(TypeName = "decimal(8,2)")]
        public decimal Price { get; set; }

        [Required(ErrorMessage = "Категорія є обов'язковою.")]
        [StringLength(50, ErrorMessage = "Категорія занадто довга.")]
        public string Category { get; set; } = string.Empty;

        public ICollection<Ticket> Tickets { get; set; } = new List<Ticket>();
    }
}

```

Висновок : Набув навичок роботи з ORM, навчився, блін, нормально робити створення, читання, оновлення та видалення даних через той клятий EF Core, а також розібрався з усією цією фігнею валідації даних.

Змн.	Арк.	№ докум.	Підпис	Дата	Арк.
					5